第四章：生成排列和组合

4.1 生成排列
4.2 排列中的逆序
4.3 生成组合
4.4 生成 *r* 子集

主要内容

- 生成组合算法
  - －压缩序
  - －反射**Gray**序

4.3 生成组合

- n元集合$S=\{x_{n-1}, \ldots, x_1, x_0\}$的所有组合（子集）共有$2^n$个。（$S$的幂集$2^S$）。
- 设计一个算法将 S 的所有组合列举出来。
  - 没有重复
- 特征函数 χ_A: $S \to \{0, 1\}$，$A \subseteq S$

$$\text{对任意} x \in S，\ \chi_A = \begin{cases} 1, & \text{若} x \in A \\ 0, & \text{若} x \notin A \end{cases}$$

注意: 长度为n的二进制数也是2^n个，两者有何联系？

- n元集合$S=\{x_{n-1}, x_{n-2}, \ldots, x_0\}$的组合与长度为$n$的二进制数一一对应

x_{n-1}	x_{n-2}	…	x_1	x_0
a_{n-1}	a_{n-2}	…	a_1	$a_0 \in \{0,1\}$

- 用二进制 $a_{n-1}a_{n-2}\ldots a_0$ 表示 S 的一个组合$\{x_{i_1}, x_{i_2} \ldots, x_{i_k}\}$，其中$n\text{-}1 \geq i_1 > i_2 > \ldots > i_k \geq 0$:

$a_{i_j} = 1$ ($j \in [1, k]$)，其他位置为0

例： $S=\{x_7, x_6, \ldots, x_1, x_0\}$的一个组合$\{x_7, x_5, x_1\}$对应的二进制数为 10100010

a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0
1	0	1	0	0	0	1	0

- n元集合$S=\{x_{n-1}, x_{n-2}, \ldots, x_0\}$的组合与长度为$n$的二进制数一一对应

x_{n-1}	x_{n-2}	…	x_1	x_0
a_{n-1}	a_{n-2}	…	a_1	$a_0 \in \{0,1\}$

- 用 S 的组合来表示$[0, 2^{n-1}]$中的 一个整数的二进制表示: 1 所在的位置对应的元素包含在组合中。

例： $n=7$，整数$29 \in [1, 2^7]$ 的二进制表示为：0011101，
29对应的组合为 $\{x_4, x_3, x_2, x_0\}$

x_6	x_5	x_4	x_3	x_2	x_1	x_0
0	0	1	1	1	0	1

- 如何生成$S=\{x_{n-1}, x_{n-2}, \ldots, x_0\}$的所有$2^n$个组合？
  - 按从小到大的顺序写出0 到 2^n-1的所有数的二进制形式
  - 每次使用二进制数的加法加 1

n=4时, $a_3a_2a_1a_0$, $\{x_3, x_2, x_1, x_0\}$的子集

0	0 0 0 0	Φ
1	0 0 0 1	x_0
2	0 0 1 0	x_1
3	0 0 1 1	x_1, x_0
4	0 1 0 0	x_2
5	0 1 0 1	x_2, x_0
6	0 1 1 0	x_2, x_1
7	0 1 1 1	x_2, x_1, x_0
8	1 0 0 0	x_3
9	1 0 0 1	x_3, x_0
10	1 0 1 0	x_3, x_1
11	1 0 1 1	x_3, x_1, x_0
12	1 1 0 0	x_3, x_2
13	1 1 0 1	x_3, x_2, x_0
14	1 1 1 0	x_3, x_2, x_1
15	1 1 1 1	x_3, x_2, x_1, x_0

$\{x_0\}$的所有组合

$\{x_1, x_0\}$的所有组合

$\{x_2, x_1, x_0\}$的所有组合

- 当$j<n-1$时，$\{x_j, \ldots, x_1, x_0\}$的所有组合都在至少含有$\{x_{n-1}, \ldots, x_{j+1}\}$中一个元素的组合的前面

——子集的压缩序

	$a_3a_2a_1a_0$
0	0 0 0 0
1	0 0 0 1
2	0 0 1 0
3	0 0 1 1
4	0 1 0 0
5	0 1 0 1
6	0 1 1 0
7	0 1 1 1
8	1 0 0 0
9	1 0 0 1
10	1 0 1 0
11	1 0 1 1
12	1 1 0 0
13	1 1 0 1
14	1 1 1 0
15	1 1 1 1

生成={$x_{n-1}, x_{n-2}, \ldots, x_0$}的所有$2^n$个组合的二进制算法

1. 初始： $a_{n-1}\ldots a_1a_0=0\ldots00$

2. 当 $a_{n-1}\ldots a_1a_0\neq 1\ldots11$时，执行以下操作：

   (1) 求出使得$a_j=0$的最小整数 j

   (2)用1替换 a_j 并用 0 替换每个$a_{j-1},\ldots,a_0$。

3. 当$a_{n-1}\ldots a_1a_0=1\ldots11$时算法结束。

二进制加法

算法按自然二进制数顺序生成， 称为n元组字典序。

问题：

- 组合$\{x_6, x_4, x_2, x_1, x_0\}$ 的下一个组合是什么？

  1010111+1=1011000，

  下一个组合为$\{x_6, x_4, x_3\}$

- 例2： $S=\{x_6, x_5 \ldots, x_1, x_0\}$的 哪个子集是子集列表中的第108个子集？

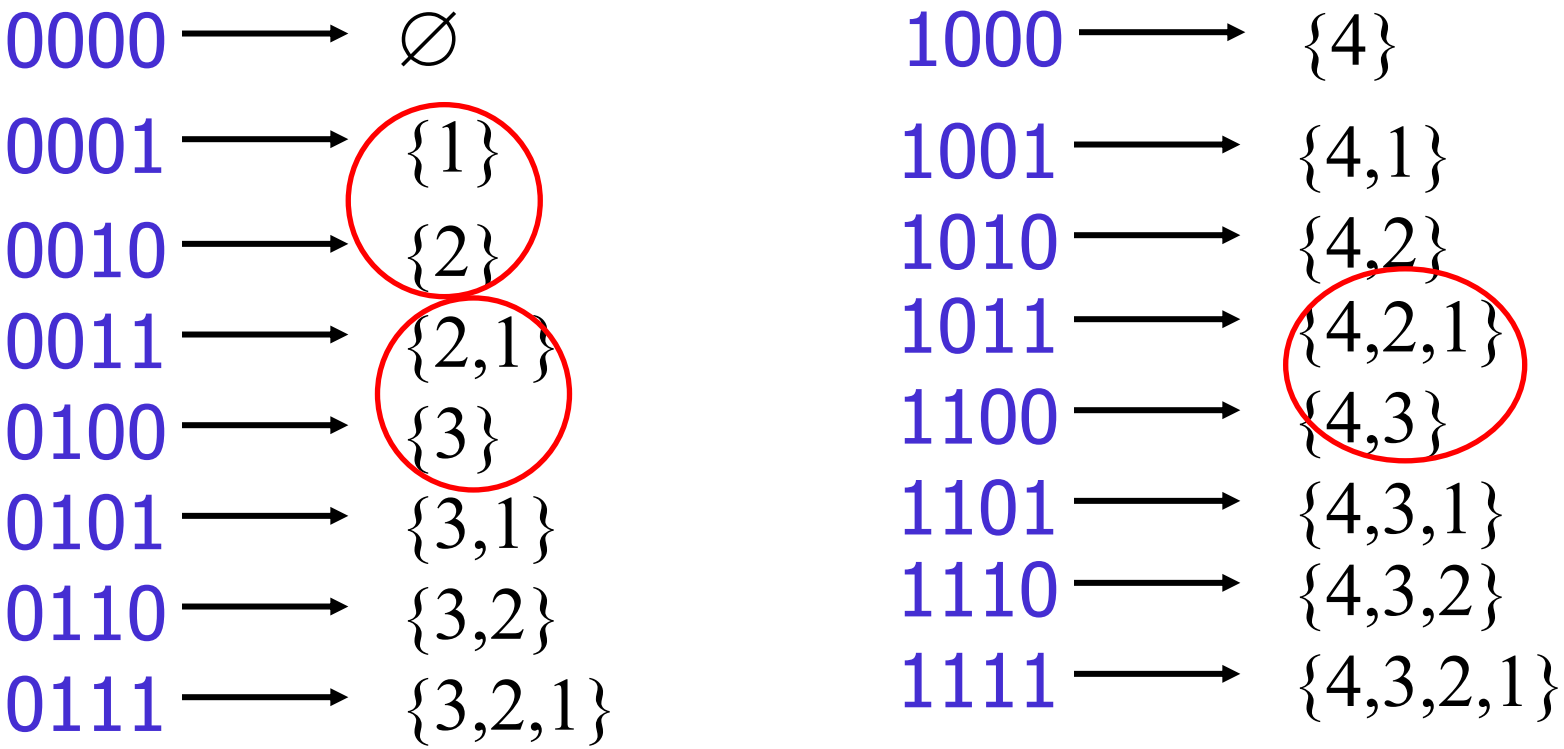
（注：列表上的位置是从 0 开始，第108个子集是指子集列表中对应108的子集）

108的二进制数：1101100

第108个子集为 $\{x_6, x_5, x_3, x_2\}$

字典序对应的组合生成

- 例：集合S={4, 3, 2, 1}的组合生成。



相邻组合可能相差较大

是否可以使得相邻的组合尽可能相似？

算法2：反射Gray码序列生成算法

- 特点：相邻的组合**仅相差一个元素**

  （增加一个或者删除一个元素）

- 如：n（=1, 2, 3）元集的组合

n=1, $\varnothing$, $\{x_0\}$

0 1

n=2, $\varnothing$, $\{x_0\}$, $\{x_1, x_0\}$, $\{x_1\}$

0 0 0 1 1 1 1 0

n=3

000 $\varnothing$

001 $\{x_0\}$

011 $\{x_0, x_1\}$

010 $\{x_1\}$

110 $\{x_2, x_1\}$

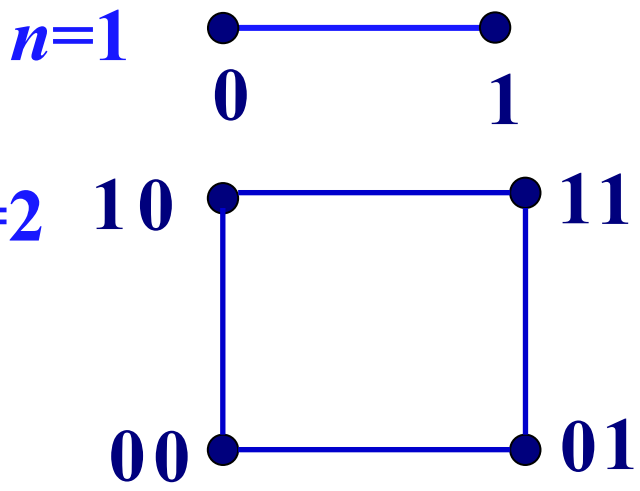
111 $\{x_2, x_1, x_0\}$

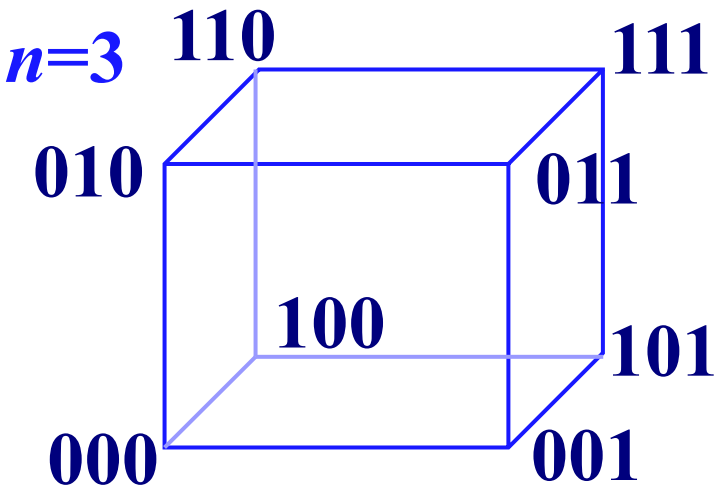
101 $\{x_2, x_0\}$

100 $\{x_2\}$

几何表示（Gray序）

- *n* 元组看作是 *n* 维空间的点的坐标　（单位 *n*方体）
- 每两个点的坐标仅有一个位置不同时，有一条连线

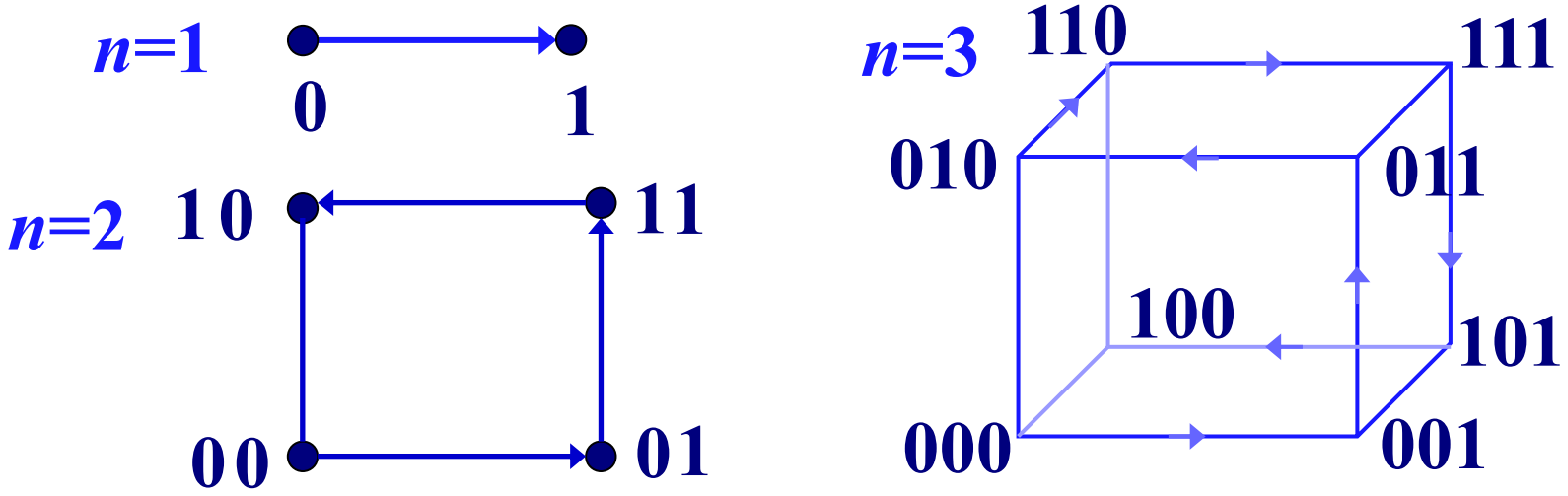




- 算法生成所有的 *n*元组：
  - ✓ 遍历 *n* 维空间的每个点，使得每个点与其后继只在一个位置不同；
  - ✓ 产生的路径称为***n*阶Gray码**

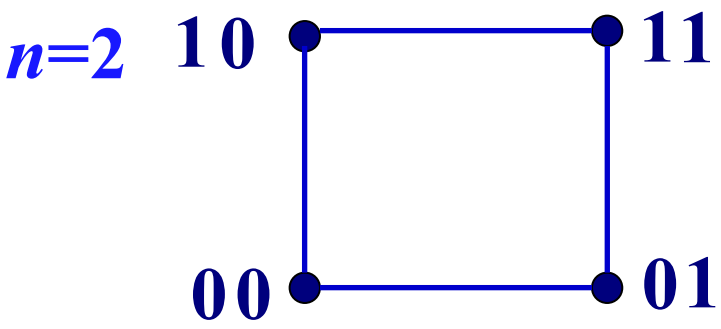
几何表示（Gray序）

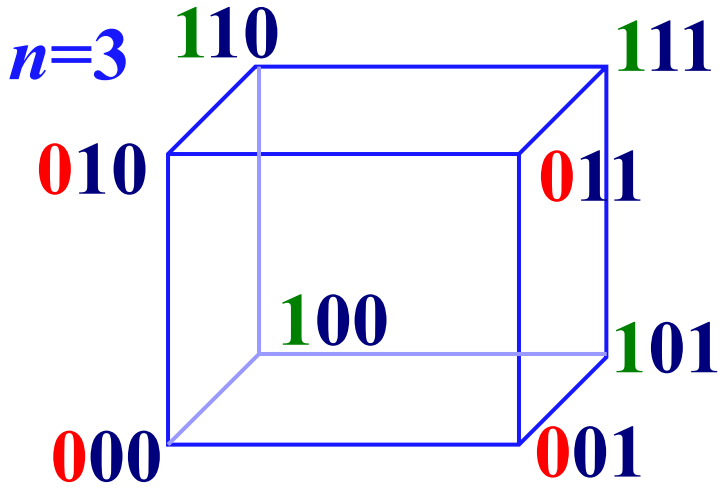
- *n* 元组看作是 *n* 维空间的点的坐标　（单位 *n*方体）
- 每两个点的坐标仅有一个位置不同时，有一条连线

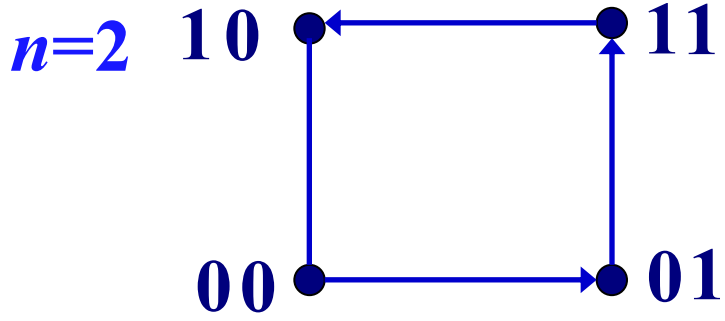


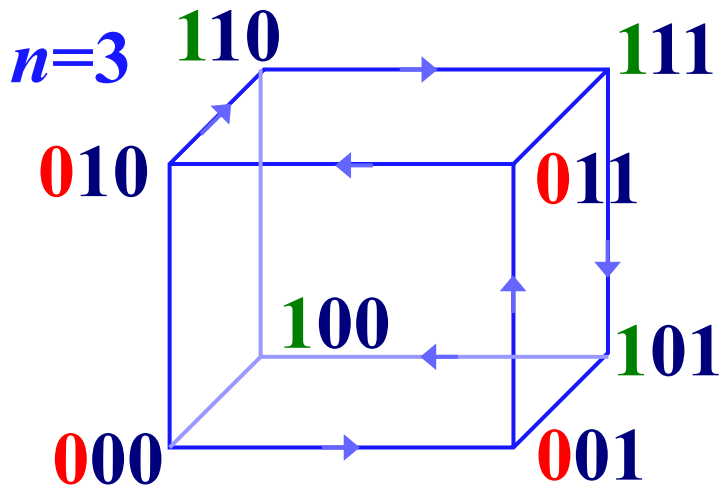
- 算法生成所有的 *n*元组：
  - ✓ 遍历 *n* 维空间的每个点，使得每个点与其后继只在一个位置不同；
  - ✓ 产生的路径称为*n*阶Gray码

遍历可以再经过一条边从终点返回到起点：循环Gray码



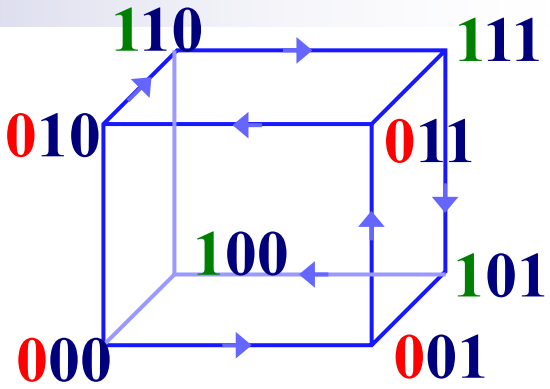






递归构造 n 阶Gray码（$n \geq 1$）：反射Gray码

n 阶Gray反射码的归纳定义



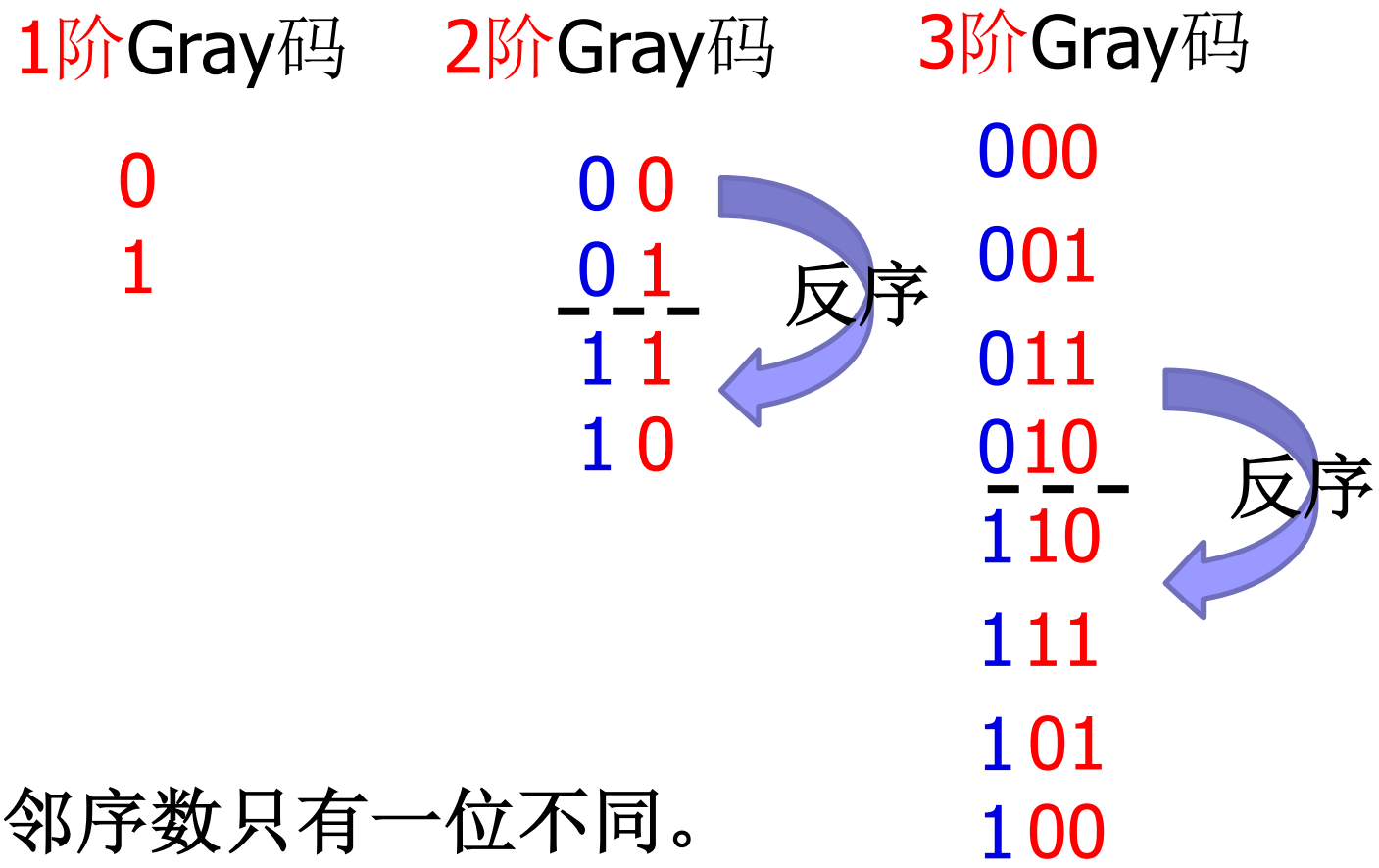
1. 1阶反射Gray码是 $\begin{matrix}0\\1\end{matrix}$ ；

2. 设n>1且n–1阶反射Gray码已经构造，如下构建 n阶反射Gray码：

(1) 以n–1阶反射Gray码所给出的顺序列出 0 和 1 的 n–1元组，把 0 添到每个 n–1 元组的开头，

(2) 再反序列出n–1阶反射Gray码的全部 n–1元组，并把1加到全部 n–1元组的开头。

- n 阶反射Gray码以 00…0开始，并以 10…0结束。
- 因为00…0 与10…0只相差一位，因此该码是循环码。



- 相邻序数只有一位不同。
- 递归方法构造反射Gray码，生成组合。

问题：能否有直接的方法构造 *n* 阶反射Gray码？

以反射Gray码的顺序直接生成0,1的n元组

1.初始：$a_{n-1}\ldots a_1a_0=0\ldots00$

2. 当$a_{n-1}\ldots a_1a_0\neq10\ldots0$ 时，进行以下操作：

(1) 计算$\sigma(a_{n-1}\ldots a_1a_0)=a_{n-1}+\ldots+a_1+a_0$

(2) 如果$\sigma(a_{n-1}\ldots a_1a_0)$是偶数，则改变 a_0（0变1或1变0）

(3) 否则，确定 j，使得$a_j=1$且对于所有$i<j$, $a_i=0$，
然后，改变a_{j+1}（0变1或1变0）.

称为逐次法。

每次改变均变化σ值的奇偶性

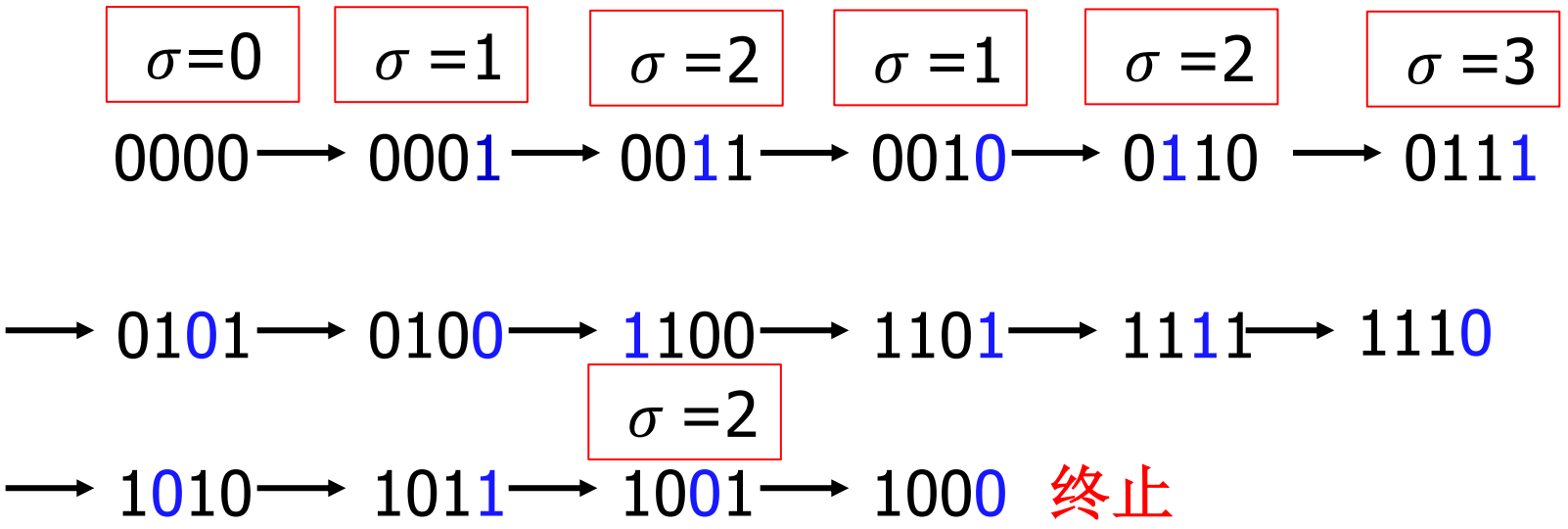
1 0 0 1 1 0 0 0 1 0 0 0

1 0 0 1 1 0 0 0 1 0 0 1

1 0 0 1 1 1 0 0 1 0 0 0

1 0 0 1 1 1 0 1 1 0 0 0

例：用逐次法生成4阶反射Gray码。



定理 4.3.1 对于每一个正整数n， 逐次法生成 n阶反射Gray码

证明：对n进行归纳证明。

1. n=1 时显然成立。
2. 假设对于n–1时，结论成立，即逐次法生成 n-1 阶反射Gray码。
3. 当对于n时，证明逐次法生成 n 阶反射Gray码。

考虑 n 阶Gray码的前 2^{n-1} 个组合与后 2^{n-1} 个组合。

0 000
0 001
0 011
0 010
0 110
0 111
0 101
0 100
1 100
1 101
1 111
1 110
1 010
1 011
1 001
1 000

(1) 考虑 n 阶Gray码的前2^{n-1}个组合：

- 是由 n-1 阶Gray码在开头添加0形成，因此不会改变σ 值的奇偶性。
- 除第 2^{n-1}个元组（010…0）外，其余元组首位的 0不影响逐次法的应用，即逐次法用于前2^{n-1}–1个元组，与逐次法生成n-1阶Gray码的顺序一致。

由归纳假设，逐次法可生成前一半的n阶Gray码。

对 n 阶反射码的第2^{n-1}个元组 (010…0)，运用逐次算法：σ(010…0)＝1，则得到 (110…0) 正好是2^{n-1}+1个 n 阶反射Gray码。

0	000
0	001
0	011
0	010
0	110
0	111
0	101
0	100
1	100
1	101
1	111
1	110
1	010
1	011
1	001
1	000

(2)考虑 n 阶Gray码的后2^{n-1}个组合：

对任意前后连续的两个n元组：

$$1a_{n-2}\ldots a_1a_0 \to 1b_{n-2}\ldots b_1b_0,$$

只需证明逐次法确实从$1a_{n-2}\ldots a_1a_0$生成$1b_{n-2}\ldots b_1b_0$。

在n-1阶反射Gray码中，有$a_{n-2}\ldots a_1a_0$在$b_{n-2}\ldots b_1b_0$之后，即 $b_{n-2}\ldots b_1b_0 \to a_{n-2}\ldots a_1a_0$ 。

由于 $\sigma(a_{n-2}\ldots a_1a_0)$ 与 $\sigma(b_{n-2}\ldots b_1b_0)$ 奇偶性相反，因此 $\sigma(1a_{n-2}\ldots a_1a_0)$ 与 $\sigma(1b_{n-2}\ldots b_1b_0)$奇偶性也相反。

(a)若$\sigma(1a_{n-2}\ldots a_1a_0)$是偶数，那么$\sigma(a_{n-2}\ldots a_1a_0)$是奇数，$\sigma(b_{n-2}\ldots b_1b_0)$是偶数，

由归纳假设，$a_{n-2}\ldots a_1a_0$由$b_{n-2}\ldots b_1b_0$改变b_0得到，

因此，由$1a_{n-2}\ldots a_1a_0$改变a_0得到$1b_{n-2}\ldots b_1b_0$，与逐次法一致。

0	000
0	001
0	011
0	010
0	110
0	111
0	101
0	100
1	100
1	101
1	111
1	110
1	010
1	011
1	001
1	000

反序

(2)考虑 n 阶Gray码的后2^{n-1}个组合：

对任意前后连续的两个n元组：

$$1a_{n-2}\ldots a_1a_0 \to 1b_{n-2}\ldots b_1b_0,$$

只需证明逐次法确实从$1a_{n-2}\ldots a_1a_0$ 生成 $1b_{n-2}\ldots b_1b_0$。

在n-1阶反射Gray码中，有$a_{n-2}\ldots a_1a_0$在 $b_{n-2}\ldots b_1b_0$ 之后，即 $b_{n-2}\ldots b_1b_0 \to a_{n-2}\ldots a_1a_0$ 。

由于 $\sigma(a_{n-2}\ldots a_1a_0)$ 与 $\sigma(b_{n-2}\ldots b_1b_0)$ 奇偶性相反，因此 $\sigma(1a_{n-2}\ldots a_1a_0)$ 与 $\sigma(1b_{n-2}\ldots b_1b_0)$奇偶性也相反。

(b) 若$\sigma(1a_{n-2}\ldots a_1a_0)$是奇数，那么$\sigma(a_{n-2}\ldots a_1a_0)$是偶数，$\sigma(b_{n-2}\ldots b_1b_0)$是奇数，

由归纳假设，$a_{n-2}\ldots a_1a_0$由$b_{n-2}\ldots b_1b_0$ 改变 b_{j+1}得到，其中 $b_j=1$，$0\le j< n-2$, 而对于所有$i<j$，$b_i=0$。

因此，由$1a_{n-2}\ldots a_1a_0$改变a_{j+1}得到$1b_{n-2}\ldots b_1b_0$，与逐次法一致。由归纳法假设，结论成立。

0	000
0	001
0	011
0	010
0	110
0	111
0	101
0	100
1	100
1	101
1	111
1	110
1	010
1	011
1	001
1	000

反序

组合的两种生成方法

2^n 个二进制 n 元组的两种线性排序

- 从00…0 开始利用二进制算术的字典序
  - 与二进制数顺序一致
- 从00…0开始的反射Gray码
  - 相邻两个子集相差一个元素
  - 递归法、逐次法

问题：如何确定 n 元组在线性排序的准确位置？

- 如何确定 n 元组在 Gray 码序表的准确位置？

给定Gray码$a_{n-1}\ldots a_1a_0$. 对于$i = 0, 1, \ldots, n-1$，设

$$b_i=\begin{cases}0, \text{若} a_{n-1}+\cdots+a_i \text{是偶数}\\ 1, \text{若} a_{n-1}+\cdots+a_i \text{是奇数}\end{cases},$$

此时，$a_{n-1}\ldots a_1a_0$ 在 Gray码序表的位置和$b_{n-1}\ldots b_1b_0$ 在字典序表上的位置相同。

即$a_{n-1}\ldots a_1 a_0$在Gray码序表的位置是

$$b_{n-1}\times 2^{n-1}+\ldots+b_1\times 2 + b_0\times 2^0。$$

《计算机程序设计艺术》第4卷2册——生成所有元组和排列，Donald E. Knuth著，苏运霖译。